

1) Suppose C is a class that implements the interfaces A and B .  
Write whether the following assignments are (V)alid or (I)nvalid statements.

```
C c = . . . ;
A i = . . . ;
B k = . . . ;
```

- a) c = i; a) \_\_\_\_\_
- b) c = (C) i; b) \_\_\_\_\_
- c) i = c; c) \_\_\_\_\_
- d) k = (C) c; d) \_\_\_\_\_
- e) i = k; e) \_\_\_\_\_
- f) k = c; f) \_\_\_\_\_  
i = (A) k;

2) Suppose C is a class that implements the interfaces A and B .  
Write whether the following assignments are (V)alid or (I)nvalid statements.

```
C c = new C ( );
A a = new C ( );
```

- a. A i = c ; a) \_\_\_\_\_
- b. B k = (B) a ; b) \_\_\_\_\_
- c. C d = (C) a ; c) \_\_\_\_\_

3) Answer (T) rue or (F)alse

- a) When casting objects, the user accepts that the cast may cause an exception. a) \_\_\_\_\_
- b) When casting numbers, the user accepts that data may be lost. b) \_\_\_\_\_

**OVER**

4) Suppose the class Sandwich implements the Edible interface.  
The Rectangle does not implement the Edible interface.  
State whether the following assignments are (V)alid or (I)nvalid statements.

a) Edible sub = **new** Sandwich( ); a) \_\_\_\_\_

b) Sandwich sub2 = **new** Sandwich( );  
Edible b = sub2; b) \_\_\_\_\_

c) Rectangle cerealBox = **new** Rectangle(5, 10, 20, 30); c) \_\_\_\_\_

d) Rectangle cerealBox = **new** Rectangle(5, 10, 20, 30);  
Edible d = cerealBox; d) \_\_\_\_\_

e) Rectangle cerealBox = **new** Rectangle(5, 10, 20, 30);  
Edible e;  
e = (Edible) cerealBox; e) \_\_\_\_\_

f) Sandwich sub = **new** Sandwich( );  
Edible f;  
sub = f; f) \_\_\_\_\_

g) Sandwich sub = **new** Sandwich( );  
Edible g;  
sub = (Sandwich) g; g) \_\_\_\_\_

h) Sandwich sub = **new** Sandwich( );  
Rectangle cerealBox = **new** Rectangle(5, 10, 20, 30);  
sub = (Sandwich) cerealBox; h) \_\_\_\_\_

## KEY

1. The following require a cast:

```
c = i; // c = (C) i;  
i = j; // i = (I) j;
```

2 None of them will throw an exception.

3.a. `Sandwich sub = new Sandwich();`

b. `Edible e = sub;`

c. `Rectangle cerealBox = new Rectangle(5, 10, 20, 30);`

e. `f = (Edible) cerealBox;` // Statement can be compiled, but gives a "ClassCastException" at runtime

g. `sub = (Sandwich) e;`